Preferences, and click on Plugins. Click on AIM/ICQ and go to Trillian Preferences again and on the right hand side click on Add a New IM Connection. Configure all of your login information for the AIM screen name that will be receiving the data from the IMfree. Login with this screen name on Trillian. For further explanation later on, my fictitious screen name will be called ItunesRemote.

5. Now for the fun stuff. I must give credit to Jason over at www.teknikill.net/bbs/ for giving me the idea and the foundation for the rest of this article. His instructions worked when I tried them, so I am only adapting them a bit to match our iTunes experiment criteria.

6. Go to Trillian -} Trillian Preferences and then click on Advanced Preferences.

7. Click on Automation in the left hand menu.

8. Click Add -} Word Matching.

9. In the Add Word Match Entry box enter the word '"launch" in the word text box, check Match Whole Word and check Generate Event, then enter something for the event type (just use "launch" again).

10. Click on Add Event.

11. Next to Action Type change Sound to Execute Program. (You probably see where this is leading to by now. If not, keep reading anyway.)

12. Browse to the location of your iTunes exe file and select it as the program that you wish to execute. It will most likely be in "C:\ Program Files\ iTunes". Click Set Event.

13. This will take you back to the Match Word Entry menu. Make sure that everything is right and that the word is "launch". Also make sure that the entry type is called "launch". Click Save.

It's now time to cook the shish kabob. Login with your IMfree screen name and IM your other screen name (ItunesRemote) with just the word "launch". Voila, iTunes launches! If your firewall is blocking iTunes from launching, just check Remember This Setting and Allow if you run on Zonealarm. Do likewise if you have a different firewall. All that you need to do to play a song, skip a song, etc. is to repeat steps 6-13 by replacing the location of the iTunes exe file with one of the iTunes scripts that was installed originally. For instance, if you wanted to play a song you would have Trillian execute the script called Play in C:\ Program Files\ iTunes\ Scripts, if that's where you put it. Also, remember to type in the word that Trillian will match with the program as Play, so that when you send the message of Play to ItunesRemote, it will execute the script and play the song.

Trillian Pro does not seem to have a limit on the number of commands that it can execute on the host PC. I have about five commands running, including the ability to change the volume, all on my IMfree. The possibilities for this application are limitless. Any application or program for that matter can be launched or executed half a world away with a cell phone. The only setback is that Trillian Pro has a price tag of $25. At least test it out with the trial version and prepare to be amazed. The IMfree can be bought on eBay for about $10 and on Amazon for $30, making this wireless iTunes remote cost between $35-$55. Imagine queueing up the song "I'll Be Home For Christmas" on your PC in America while sitting in the Tokyo airport with nothing but your cell phone. Please, let the imagination run wild.

The Not-So-Great Firewall of China

by Tokachu
tokachu@gmail.com

When most people think of Internet censorship, they tend to think about China the most. While many other countries have some sort of state-controlled Internet policy, most people would refer to China because of the sheer size of the population and government. Ironically enough, the country with one of the largest Internet populations seemed to go for the lowest bidder when it came to Internet censorship devices, replacing quality control with frantic developers pressed for time.

No matter how strange that may be, it still does not justify a government which wants to keep full control over all media. Which is why I'll tell you, and hopefully a Chinese friend, how the "Great" firewall works and how to keep it from ruining your Internet.

How It Works

Unlike most other countries that simply block all TCP traffic or utilize a filtering HTTP proxy, China relies almost solely on special routers designed to censor based on raw TCP data instead of HTTP requests. The government of China relies on two main methods of censorship: flooding fake DNS requests and forging TCP connection resets.

DNS Poisoning

Very few domain names are actually "blocked" using this method. For a DNS poison to take place, there must be a request for a very, very, very naughty website (like minghui.org) placed. This keeps anyone from figuring out how to connect to, let alone down-

load content from, a forbidden host.

Here's how an uncensored DNS request would look like in China:

```
0.000000 192.168.1.2 -> 220.194.59.17
DNS Standard query A baidu.com
0.289817 220.194.59.17 -> 192.168.1.2
DNS Standard query response A
202.108.22.33 A 220.181.18.114
```

And here's how it would look if a domain were censored:

```
0.000000 192.168.1.2 -> 220.194.59.17 DNS
Standard query A minghui.org 0.288963
220.194.59.17 -> 192.168.1.2 DNS Standard
query response A 203.105.1.21 0.289482
220.194.59.17 -> 192.168.1.2 DNS Standard
query response A 203.105.1.21 0.289838
220.194.59.17 -> 192.168.1.2 DNS Standard
query response A 203.105.1.21 0.290374
220.194.59.17 -> 192.168.1.2 DNS Standard
query response A 203.105.1.21 0.290732
220.194.59.17 -> 192.168.1.2 DNS Standard
query response A 203.105.1.21 0.290757
192.168.1.2 -> 220.194.59.17 ICMP
Destination unreachable (Port unreachable)
0.291311 220.194.59.17 -> 192.168.1.2 DNS
Standard query response A 169.132.13.103
0.291337 192.168.1.2 -> 220.194.59.17 ICMP
Destination unreachable (Port unreachable)
```

The real reply never gets through because the router forges nearly a half dozen fake DNS replies, along with a few random ICMP messages, to whoever requests a "forbidden" website. This filter only works on UDP port 53 (DNS), which would theoretically make uncensored DNS requests possible if a sufficient number of DNS servers running on ports other than 53 existed.

You can tell if your packets are going through a Chinese router by one simple test. Try performing a DNS query to a remote machine in China. If it doesn't go through, try performing a DNS query for "minghui.org" on the same machine. If you get seemingly random responses, you're routing through China. If you want to determine which router is responsible for the censorship, run a traceroute and perform DNS requests on each hop, starting at the closest. When you get the fake DNS replies, you've found the offending router.

Forging TCP Resets

If a TCP connection is made from or to a computer in China, the packet data is checked for any "forbidden" words. If the data contains any of those words, the router forges a TCP RST (reset connection) packet. This also triggers a temporary block on TCP connections between those two specific computers. This makes it appear that the server has gone down temporarily.

The list of words not permitted to be used are encoded in GB2312 format, which ensures that businesses with websites in China will not be able to send any illegal content to computers in China (since GB2312 is a character set required to be supported by all applications in China). The filter works thusly:

If the word can be written in pure ASCII, look for the word in any mixture of lowercase and uppercase ASCII letters.

If the word must be written in any combination of CJK ideographs, look for the byte sequence in either raw or URL-encoded GB2312. Hexadecimal strings are also case-insensitive.

Problems

Nearly all the problems of China's firewalls stem from one problem with the routers: they all perform stateless packet inspection. It doesn't matter what protocol the packets are using, nor what computer a packet comes from. All the router is concerned with is finding packets and forging responses, not dropping content.

Unfortunately, that flaw puts the router owners and admins at an extreme disadvantage. Anybody can do a Google search for packet-forging software or libraries (such as libpcap) and whip up a script to flood Chinese routers with fake packets - and the routers will respond, no matter what. It wouldn't be difficult to set up a botnet with DNS request forgers that can send billions of fake DNS requests to various routers, and in return have the victim think China is attacking his or her server! It's also possible to forge a TCP data packet with fake source and destination addresses, which means that if you happened to know the IP addresses of two important diplomats, you could easily cut off their ability to communicate. Popular Chinese websites are just as vulnerable too; email systems could be cut off for hours at a time. The possibilities are endless. The TCP RST timer may be fairly short, but keep in mind that it only takes one fake packet to close a connection.

Getting Around It

The TCP Stack. One way to tell fake RST packets from real RST packets is to look at the time-to-live (TTL) parameter. Forged packets will always have higher TTLs than the real ones. Getting around this, however, would require that both parties have a stateful TTL comparison filter at the kernel level. That's no good.

You could, however, rewrite a TCP-based application to send "forbidden" words by using the TCP urgent flag (URG). This only requires that both parties have a modified application - no kernel tweaking necessary. A great example of a program that sends data like that is a proof-of-concept C program called "covertsession" (search for it on Packet Storm Security). It can bypass most stateful packet inspectors, so it easily gets around the stateless inspectors in China. This is probably the best way to modify instant messaging (such as QQ) and IRC applications, assuming one couldn't just use encryption on both ends.

HTTP Traffic. There's nothing really special about how the firewall treats HTTP traffic. Mind you that it only looks for certain strings, no matter where they are. But notice how I said it only uses the GB2312 character set: there's nothing stopping us from simply using UTF-8 instead. You can "switch" your websites from GB2312 to UTF-8 by simply running them through iconv. It's impossible for any UTF-8 sequence to match a GB2312 sequence, even by

accident, so you're partially assured good exposure (for a period of time).

Most China-based web hosts, such as Baidu and Yahoo! China, rely on the firewalls to block some content for them. Google China, however, is the one huge exception. Google's Chinese servers are located in the United States and their censorship is done entirely in-house. What does that mean? For one, we don't need to worry about text being sent in GB2312 format (Google insists on using UTF-8). We can also exploit a "feature" in Google's text engine that was overlooked during the Google China development.

Google doesn't compare strings in their text engine like most of us do. Instead of simply comparing bytes, Google considers some words and characters equal to other words and characters that wouldn't match with a byte comparison algorithm. The character equality is what we want to look at here: mainly, how Google considers "fullwidth" ASCII characters (wide, fixed-width characters mostly used in Japanese character sets) equal to their ASCII counterparts. If you were to search for "computers" using fullwidth characters, you'd get the same results as you would with a simple ASCII search (although some ads might not show up).

Now here's where the hack comes in: Google's censors don't look for those fullwidth characters. So, if we were to search Google China for "tiananmen square" using fullwidth characters, the results wouldn't be filtered (the connection may be reset from what Google sends). Luckily, this trick works for Google Images - meaning that it isn't too hard to get Google's cache of images normally unfindable in China!

Here's some sample code to generate fullwidth characters in a shell in Perl (assuming you've got Unicode support in your terminal):

```
#!/usr/bin/perl -w
# fw.pl - make text W-I-D-
E (convert ascii to fullwidth)
use encoding "UTF-8";
$input = $ARGV[0] or die("need
one argument for text");
foreach (split //, $input) { print
chr(0xFEE0 + ord($_)); }
## end script
```

Just type whatever search term you want, plug in the output to Google, and watch once-censored search results just show up!

Conclusion

Censorship isn't a profitable business. If China were to release an honest budget (and if people and corporations found out a huge percentage of their GDP was going towards censorship and propaganda instead of food and health care), China's economy would collapse in a matter of hours. Sadly, it isn't just Chinese citizens who believe the lies: corporations like Cisco and Google actually believe you can make money by keeping information from people. The sooner the Chinese people and their government realize this, the better.

(There are far too many people to thank - you know who you are.)

Hactivism in the Land Without a Server

by \ /indic8tr

A little while back I stumbled upon a link to the forums of the Korean Friendship Association (http://➥www.korea-dpr.com/cgi-bin/simpleforum.cgi). Naturally, I thought they needed to hear my opinion on the plight of the people of North Korea. Unfortunately, there is no obvious way of registering for a forum membership without joining their club, nor could I discover any less obvious means to gain access.

Not being content to walk away in total defeat, I decided to examine other parts of the site. After a little research, I discovered that this domain in fact houses the official website of the Democratic People's Republic of Korea. A whois search for the korea-dpr.➥com domain shows that the server is located in, of all places, Spain.

This seems counterintuitive at first glance. However, this makes perfect sense for a country where information is so tightly controlled that it is a capital crime to own a radio that is not hardwired to receive only the single government-approved station. That the DPRK cannot permit their own government's public website, their equivalent to whitehouse.gov, to be located on a server within its own borders flows naturally from this mindset. Clearly, North Korea isn't a place that is easily targeted by those who would seek to use online activism to further the free flow of knowledge. This is frustrating, because hactivism is one of the few nonviolent routes we have to bring the fight to those who would stifle learning and creativity both at home and abroad.

While we can't pick on Dear Leader directly, someone could hypothetically stick it to his fan club. Using techniques similar to the "Having Fun with Cookies" article in 23:3, a malicious user can use inline javascript in a browser's address bar to get free stuff courtesy of the Korean Friendship Association.